In the Claims:

Claim 1 (currently amended): A computer implemented method for unified simulation of a system design and an application program, comprising:

identifying a memory component and a processor component of said system design;

identifying a software component of said application program;

creating a programming model of said memory component and said processor component;

associating said software component with said programming model of said memory component within a simulation environment; and

executing said system design and said application program in said simulation environment whereby said software component is simulated in association with said memory component and said processor component;

running said unified simulation as a single process and allowing said application program to be loaded into said memory component for simulation.

Claim 2 (original): The computer implemented method of claim 1 whereby the executing said system design and

said application program in said simulation environment step further comprises:

generating cycle accurate information for said system design in said simulation

environment.

Claim 3 (original): The computer implemented method of claim 1 further comprising:

creating a set of function calls in a high level general purpose programming language;

linking said memory component with a processor component wherein said link comprises:

at least one of said set of function calls: and

executing said system design and said application program in said simulation environment.

Claim 4 (original): The computer implemented method of claim 3 wherein said executing step further comprises:

reading said software component from said memory component into said processor component through said link;

processing instructions of said software component on said processor component during said simulation; and

generating cycle accurate information in said simulation environment.

Claim 5 (original): The computer implemented method of claim 4 wherein said

generating step further comprises:

dividing activities in said simulation environment into a first plurality of activities comprising an execute phase and a second plurality of activities comprising an update phase;

computing said first plurality of activities comprising said execute phase at a clock edge of a virtual system clock;

updating at said clock edge a state of said simulation environment; and

computing said second plurality of activities comprising said update phase at said clock edge.

Claim 6 (original): The computer implemented method of claim 1 wherein said programming model is created in a high level general purpose programming language.

Claim 7 (currently amended): A computer implemented method for simulating a digital system design and an application program in a single simulation environment comprising:

creating a system design model in a high level general purpose programming language,

wherein said system design model is comprised of at least a processor component and a memory component;

operatively associating said application program with said memory component;

executing said system design model and said application program in said single simulation environment wherein said processor component simulates the operation of said application program; and

maintaining cycle accurate information during the simulation;

running a unified simulation as a single process and allowing said application program to be loaded into said memory component for simulation.

Claim 8 (original): The computer implemented method of claim 7 wherein said operatively associating step further comprises:

creating a set of function calls in a high level general purpose programming language; and

linking said memory component with said processor component wherein said link comprises at least one of said set of function calls.

Claim 9 (original): The computer implemented method of claim 8 wherein said executing step further comprises:

reading said software component from said memory component into said processor component through said link; and

processing instructions of said software component on said processor component during said simulation.

Claim 10 (original): The computer implemented method of claim 7 wherein said maintaining step further comprises:

dividing activities in said simulation environment into a first plurality of activities and a second plurality of activities;

computing said first plurality of activities at a clock edge;

updating at said clock edge a state of said single simulation environment; and

computing said second plurality of activities at said clock edge.

Claim 11 (currently amended): A computer program product for simulating hardware and software components in a single simulation environment, said computer program product comprising:

instructions for identifying a memory component, a processor component, and at least one software component comprised of computer program instructions;

instructions for creating a model of said memory component and said processor component in a high level general purpose programming language;

instructions for creating a set of function calls in a high level general purpose programming language;

instructions for communicatively linking said models of said memory component and said processor component wherein said link comprises at least one of said set of function calls;

instructions for operatively associating said at least one software component with

said model of said memory component; and

instructions for executing said components in said single simulation environment

whereby said model of said processor component reads said at least one software component from said model of said memory component through said link and said model of said processor component performs said computer program instructions contained within said software component;

wherein said computer program product further includes instructions for running a unified simulation as a single process and allowing an application program to be loaded into said memory component for simulation.

Claim 12 (currently amended): A method for simulating a design and a program, comprising:

creating a memory model and a processor model;

associating a software component with said memory model within a simulation environment; and

executing said design and said program in said simulation environment whereby said software component is simulated in association with said memory model and said processor model;

running a unified simulation as a single process and allowing said program to be loaded into said memory component for simulation.

Claim 13 (original): The method of claim 12 wherein said simulation environment is an HDL simulation environment and said executing step generates cycle accurate information.

Claim 14 (original): The method of claim 13 wherein said software component contains a set of instructions and said executing step further comprises said processor model executing said set of instructions of said software component associated with said memory model.

Claim 15 (currently amended): A computer program product for simulating a design and a program, comprising:

instructions for instantiating a memory model and a processor model in a simulation environment;

instructions for associating a software component with said memory model within said simulation environment; and

instructions for executing said design and said program in said simulation environment whereby said software component is simulated in association with said memory model and said processor model;

wherein said computer program product further includes instructions for running a unified simulation as a single process and allowing said program to be loaded into said memory component for simulation.

Claim 16 (original): The computer program product of claim 15 wherein said simulation environment is an HDL simulation environment and said instructions for executing generate cycle accurate information.

Claim 17 (original): The computer program product of claim 16 wherein said instructions for executing further comprise instructions for said processor model to execute instructions of said software component associated with said memory model.